

that the file will not be deleted upon closing and said second value indicates that the file will be deleted upon closing, the method further comprising:

receiving a delete-on-close request from said client;
 changing said delete-on-close status from said first value to said second value; and
 deleting the file upon closing.

10. (Amended Twice) A method for creating a secure file on a file system of a router, said router further including a request handler and a verification routine, said method comprising:

receiving from a user at said request handler an open for write call for a file that does not exist at the time said call is received;

recognizing at said request handler that said file does not exist at the time said call is received;

creating with said request handler a file entry for said file;

receiving from said user at said request handler an authorization credential;

authenticating with said verification routine the privileges of said user;

recognizing with said request handler a combination of said user sending an open for write call for a file that does not exist at the time said call is received and said authorization credential that is authenticated; and

creating in said file system said secure file having a fixed file security status being of a first type.

11. (Amended Once) The method of claim 10, further comprising:

B5
setting a memory location associated with said file and in said file system of said router to a value indicating that said file is a secure file.

13. (Amended Twice) A program storage device readable by a machine, tangibly embodying a program of instructions executable by the machine to perform a method for creating a secure file on a file system, the program of instructions including a request handler routine and a verification routine, the method comprising:

receiving from a user with said request handler routine an open for write call for a file that does not exist at the time said call is received;

B6
recognizing with said request handler routine that said file does not exist at the time said call is received;

creating with said request handler routine a file entry for said file;

receiving from said user with said request handler routine an authorization credential;

authenticating with said verification routine the privileges of said user;

recognizing with said request handler routine a combination of said user sending an open for write call for a file that does not exist at the time said call is received and said authorization credential that is authenticated; and

creating said secure file having a fixed file security status being of a first type.

Please add claims 14-26 as follows:

B7
14. (New) The program storage device as defined in claim 13, wherein the method further comprises:

setting a memory location associated with said file to a value indicating that said file is a secure file.

15. (New) The program storage device as defined in claim 13, wherein the method further comprises:

closing said file entry.

16. (New) A router for creating a secure file on a file system of said router, said router comprising:

means for receiving from a user an open for write call for a file that does not exist at the time said call is received;

means for recognizing that said file does not exist at the time said call is received;

means for creating a file entry for said file;

means for receiving from said user an authorization credential;

means for authenticating the privileges of said user;

means for recognizing a combination of said user sending an open for write call for a file that does not exist at the time said call is received and said authorization credential that is authenticated; and

means for creating in said file system said secure file having a fixed file security status being of a first type.

17. (New) The router as defined in claim 16, further comprising:

means for setting a memory location associated with said file and in said file system of said router to a value indicating that said file is a secure file.

18. (New) The router as defined in claim 16, further comprising:

means for closing said file entry.

19. (New) A router for creating a secure file on a file system of said router, said router comprising:

a request handler wherein:

an open for write call is received from a user for a file that does not exist at the time said call is received,

a recognition is made that said file does not exist at the time said call is received,

a file entry is created for said file,

an authorization credential is received from said user,

B1
a recognition is made that a combination of said user sending an open for write call for a file that does not exist at the time said call is received and said authorization credential that is authenticated exists, and

said secure file having a fixed file security status being of a first type is created in said file system;

and

a verification routine wherein the privileges of said user are authenticated.

20. (New) The router as defined in claim 19, wherein said request handler sets a memory location associated with said file and in said file system of said router to a value indicating that said file is a secure file.

21. (New) The router as defined in claim 19, wherein said request handler closes said file entry.

22. (New) The program storage device as defined in claim 9, wherein the apparatus has a third memory associated with said file, said third memory storing a delete-on-close status, said third memory initially storing a first value and changeable to a second value wherein said first value indicates that said file will not be deleted upon closing and said second value indicates that said file will be deleted upon closing, and wherein the method further comprises:

receiving a delete-on-close request from said client;

changing said delete-on-close status from said first value to said second value; and

deleting the file upon closing.

B⁷
23. (New) An apparatus for controlling operations by a client on a stored file, said apparatus comprising:

a first memory associated with said file, said first memory for storing a fixed file security status of a first type;

a second memory associated with said file, said second memory capable of storing an active file security status of said first type and changeable to a second type wherein said first type indicates that operations are not allowed on said file and said second type indicates that operations are allowed on said file;

an independent verification routine having access to a security database listing clients and their corresponding privileges and having receipt of an authorization credential from said client;

means for copying said first type from said fixed file security status stored in said first memory to said active file security status stored in said second memory; and

means for changing said active file security status stored in said second memory to said second type if said independent verification routine determines that said client has the privilege to access said file.

24. (New) The apparatus as defined in claim 23, wherein said apparatus further comprises:

a third memory associated with said file, said third memory for storing a delete-on-close status, said third memory initially storing a first value and changeable to a second value wherein said first value indicates that said file will not be deleted upon closing and said second value indicates that said file will be deleted upon closing;

means for receiving a delete-on-close request from said client;

means for changing said delete-on-close status from said first value to said second value;

and

means for deleting said file upon closing.

25. (New) The apparatus as defined in claim 24, wherein said first memory is an NVRAM and said second memory and said third memory are in a file entry.

26. (New) The apparatus as defined in claim 24, wherein said first memory, said second memory, and said third memory comprise single bits.
